

Abstract

A method of, and apparatus for, interfacing the hardware of a processor capable of processing instructions from more than one type of instruction set. More particularly, an engine responsible for fetching native instructions from a memory subsystem (such as an EM fetch engine) is interfaced with an engine that processes emulated instructions (such as an x86 engine). This is achieved using a handshake protocol, whereby the x86 engine sends an explicit fetch request signal to the EM fetch engine along with a fetch address. The EM fetch engine then accesses the memory subsystem and retrieves a line of instructions for subsequent decode and execution. The EM fetch engine sends this line of instructions to the x86 engine along with an explicit fetch complete signal. The EM fetch engine also includes a fetch address queue capable of holding the fetch addresses before they are processed by the EM fetch engine. The fetch requests are processed such that more than one fetch request may be pending at the same time. If a pending fetch request is canceled due to a pipeline flush, then the fetch address queue is cleared and the pending fetch requests are canceled. The system also prevents macroinstruction (MIQ)-related stalls by using a speculative write pointer to control the issuance of fetch requests, thereby preventing the MIQ from becoming oversubscribed.